HN

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/963,659	09/27/2001	Ivan Teblyashkin	550-272	9245

7590 04/05/2005

NIXON & VANDERHYE P.C.
8th Floor
1100 North Glebe Road
Arlington, VA 22201-4714

EXAMINER	
SCHUBERT, KEVIN R	

ART UNIT	PAPER NUMBER
2137	

DATE MAILED: 04/05/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/963,659	TEBLYASHKIN, IVAN	
	Examiner	Art Unit	
	Kevin Schubert	2137	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *27 September 2001*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-36* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-36* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *07232002*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Claims 1-36 have been considered.

Title

The title of the invention, "Computer Virus Detection", is not descriptive. The examiner suggests the title be changed to "Computer Virus Detection Based on a Threshold of Suspect Instructions". The title change is suggested but not required.

Claim Objections

Claims 12,24, and 36 are objected to because of the following minor informalities: "exceed" should be "exceeded". Appropriate correction is required.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Claims 1-2,4-5,8-9,12-14,16-17,20-21,24-26,28-29,32-33, and 36 are rejected under 35 U.S.C. 102(e) as being anticipated by Yann, U.S. Patent Application Publication No. 2002/0078368.

As per claims 1,13, and 25, the applicant describes a method of detecting a computer virus comprising the following limitations which are met by Yann:

a) analysis logic operable to analyse program instructions forming said executable computer program to identify suspect program instructions forming said executable computer program to identify suspect program instructions being one or more of:

i) a program instruction generating a result value not used by another portion of said executable computer program;

ii) a program instruction dependent upon an uninitialised variable ([0030]);

b) detecting logic operable to detect said executable computer program as containing a computer virus if a number of suspect program instructions identified for said executable computer program exceeds a threshold level ([0030]; [0016]);

The applicant should note that both i) and ii) are disclosed.

As per claims 2,14, and 26, the applicant describes the computer program product of claims 1,13, and 25, which are met by Yann (see above), with the following limitation which is also met by Yann:

Wherein said computer virus is a polymorphic computer virus ([0030]);

As per claims 4,16, and 28, the applicant describes the computer program product of claims 1,13, and 25, which are met by Yann (see above), with the following limitation which is also met by Yann:

Wherein for each program instruction said analysis logic is operable to make a determination as to which state variables are read by that program instruction ([0034]);

The applicant should note that the variables read by the program instruction are labeled as being in a "used" state. The applicant should also note that program instructions are evaluated on a one-by-one basis ([0031]).

As per claims 5,17, and 29, the applicant describes the computer program product of claims 1,13, and 25, which are met by Yann (see above), with the following limitation which is also met by Yann:

Wherein for each program instruction said analysis logic is operable to make a determination as to which state variables are written by that program instruction ([0035]);

The applicant should note that the variables written by the program instruction are labeled as being in a "set" state.

As per claims 8,20, and 32, the applicant describes the computer program product of claims 1,13, and 25, which are met by Yann (see above), with the following limitation which is also met by Yann:

Wherein said analysis logic is operable to maintain an initialization table indicating which state variables have been initialized ([0033],[0034], and 24 of Fig 2);

Uninitialized variables are labeled "undefined". Initialized variables are labeled as "set" if written to or "used" if in a used state.

As per claims 9,21, and 33, the applicant describes the computer program product of claims 8,20, and 32, which are met by Yann (see above), with the following additional limitation which is also met by Yann:

Wherein a state variable is marked as initialized upon occurrence of any one of:

(i) a write to said state variable of a determined initialized value; and

(ii) use of said state variable as a memory address value by a program instruction ([0034]).

As per claims 12,24, and 36, the applicant describes the computer program product of claims 1,13, and 25, which are met by Yann (see above), with the following additional limitation which is also met by Yann:

Wherein if said threshold level is exceed, then further virus detection mechanisms are triggered to confirm the presence of a computer virus ([0016]).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 3,6-7,15,18-19,27, and 30-31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Yann.

As per claims 3,15, and 27, the applicant describes the computer program product of claims 1,13, and 25, which are met by Yann (see above), with the following limitation which is also met by Yann:

Wherein said analysis logic is operable to maintain a dependence table indicating dependency between state variables within said computer and loaded variable values ([0030]);

Yann discloses all the limitations of the independent claims. Yann also discloses the idea of claims 3,15, and 27 but fails to disclose the use of an actual table.

As disclosed by applicant, the dependency table is used to keep track of which state variables the program instructions read and write to in order to draw up a dependency used to indicate redundant code in the program (page 3). The "redundant code will typically be program instructions producing result values that are not used by any other portion of the computer program" (page 3).

Yann discloses that operand values, or variable values loaded into registers, are monitored to detect when an operand value is not used by the instruction ([0017]). Thus the goal of the dependency table of the applicant is met by Yann's system of keeping track of operand values which are loaded into registers, or state variables, in order to detect suspicious, unused values.

Claims 3,15, and 27 are rejected under U.S.C. 103(a) instead of U.S.C. 102(e) because Yann's system never discloses the use of an orderly dependence table to store the state variables, or registers, and their dependency on particular variables. However, since Yann does disclose monitoring the registers and their loaded variable values, or operands, ([0030]) it would have been obvious to one of ordinary skill in the art at the time the invention was filed place the data in a stored table for ease of monitoring.

As per claims 6,18, and 30, the applicant describes the computer program product of claims 3,15, and 27, which are met by Yann (see above), with the following limitation which is also met by Yann:

Wherein for each program instruction said analysis logic is operable to make a determination as to which state variables are read and written by that program instruction and for each loaded variable value within said dependence table if any state variable read by that program instruction is marked as dependent upon said loaded variable value, then all state variables written by that program instruction are marked as dependent upon said loaded variable value with previous dependencies being cleared ([0034]);

The analysis logic is operable to make a determination as to which state variables are read (in a "used" state) or written (in a "set" state).

The second part of the claim beginning with "for each loaded variable" discloses the idea that when a loaded value is loaded to a variable or variables with prior dependencies, the prior dependencies are cleared and the variable or variables are marked as dependent on the new loaded value. It is inherent that a loaded value changes the dependent variable or variables from its previous dependent value to the loaded value. For example, if registers a and b were dependent on the value 5 at location FF in memory, registers a and b will be dependent on value 8 if the value at FF is changed from 5 to 8.

The use of marking the limitation as dependent on the new value (8 in the example above) is lacking in Yann because Yann lacks an actual table for monitoring the registers or state variables and the loaded values. Furthermore, claims 6,18, and 30, are rejected on the same grounds as 3,15, and 27 because an actual table would be an obvious improvement on Yann's system because Yann already discloses monitoring the dependencies between the operands, or loaded values, and the state variables, or registers, just not in a table fashion.

As per claims 7,19, and 31, Yann discloses the computer program product of claims 3,15, and 27, which are met by Yann (see above), with the following additional limitation which is also met by Yann:

Wherein said state variables include one or more of:

(i) register values;

(ii) processing result flag values;

(iii) a flag indicative of a write to a non-register storage location ([0034]).

Claims 10-11,22-23, and 34-35 are rejected under 35 U.S.C. 103(a) as being unpatentable over Yann in view of Nachenberg, U.S. Patent No. 6,357,008.

As per claims 10,22, and 34, the applicant describes the computer program product of claims 1,13, and 25, which are met by Yann (see above), with the following limitation which is met by Nachenberg:

Wherein said analysis logic is operable to parse said executable computer program for suspect program instructions by following execution flow and upon occurrence of a branch first following a first branch path having saved pending analysis results and subsequently returning to follow a second branch path having restored said pending analysis results (Nachenberg: Col 10, line 3 to Col 11, line 25);

Yann discloses all the limitations of the independent claims. However, Yann fails to disclose the method of following each branch of a branching point to determine suspicious behavior.

Nachenberg discloses a polymorphic virus detection system which is similar to that of Yann and the applicant. Nachenberg also discloses the idea of executing each branch of a virus detection system because "infectious code may remain in an untaken branch" (Col 10, lines 34-35).

It would have been obvious to one of ordinary skill in the art at the time the invention was filed to incorporate the ideas of Nachenberg with those of Yann and add a method to explore infectious code in an untaken branching point as a further security measure to examine whether code is viral.

As per claims 11,23, and 35, the applicant describes the method of claims 10,22, and 34, which are met by Yann in view of Nachenberg (see above), with the following limitation which is met by Nachenberg:

Wherein a branch path stops being followed when any one of:

(i) there are no further suitable program instruction for execution within that branch path; and

(ii) said branch path rejoins a previously parsed execution path (Col 10, lines 26-36);

It is inherent in Nachenberg that the entire branch path is followed to detect infectious code and the branch path stops being followed when there are no further suitable program instruction for execution within that branch path.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kevin Schubert whose telephone number is (571) 272-4239. The examiner can normally be reached on M-F 8:00-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Andrew Caldwell can be reached on (571) 272-3868. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Andrew Caldwell

ANDREW CALDWELL
SUPERVISORY PATENT EXAMINER

***